I Erläuterungen

Voraussetzungen gemäß KCGO und Abiturerlass in der für den Abiturjahrgang geltenden Fassung

Standardbezug
Die nachfolgend ausgewiesenen prozessbezogenen Kompetenzbereiche sind für die Bearbeitung der jeweiligen Aufgabe besonders bedeutsam. Darüber hinaus können weitere, hier nicht ausgewiesene prozessbezogene Kompetenzbereiche für die Bearbeitung der Aufgabe nachrangig bedeutsam sein, zumal die Kompetenzbereiche in engem Bezug zueinanderstehen. Die Operationalisierung des Standardbezugs erfolgt in Abschnitt II.

Aufgabe	Prozessbezogene Kompetenzbereiche				
	P1	P2	P3	P4	P5
1.1		X			
1.2		X	X		
1.3		X			
2		X	X		
3.1			X	X	
3.2					X
4.1		X			
4.2	X				X
4.3		X	X		
4.4		X			
4.5	X				X
5			X		

Inhaltlicher Bezug
Der vorliegende Vorschlag bezieht sich schwerpunktmäßig auf die inhaltsbezogenen Kompetenzbereiche Formale Sprachen und Automaten (I2), Information und Daten (I3) und Informatiksysteme (I4) nach KCGO.

Q3: Konzepte und Anwendungen der theoretischen Informatik
verbindliche Themenfelder: Endliche Automaten (Q3.2); Formale Sprachen und Grammatiken (Q3.3); Registermaschine (Q3.5)

II Lösungshinweise und Bewertungsraster

In den nachfolgenden Lösungshinweisen sind alle wesentlichen Gesichtspunkte, die bei der Bearbeitung der einzelnen Aufgaben zu berücksichtigen sind, konkret genannt und diejenigen Lösungswege aufgezeigt, welche die Prüflinge erfahrungsgemäß einschlagen werden. Lösungswege, die von den vorgegebenen abweichen, aber als gleichwertig betrachtet werden können, sind ebenso zu akzeptieren.

Aufg.	erwartete Leistungen	BE
1.1	sechs Rumpfteile: *S2R1R2R3R3R2R1T4* sieben Rumpfteile: *S2R1R2R3R4R3R2R1T4*	1

Nicht für den Prüfling bestimmt

Aufg.	**erwartete Leistungen**	**BE**
1.2	T = {S, R, T, 1, 2, 3, 4} N = {A, Rumpf, X} Startsymbol = A P = {A → S X Rumpf T X, X → 1 | 2 | 3 | 4, Rumpf → R 1 | R 2 | R 3 | R 4 | R 1 R 1 | R 2 R 2 | R 3 R 3| R 4 R 4 | R 1 Rumpf R 1 | R 2 Rumpf R 2 | R 3 Rumpf R 3 | R 4 Rumpf R 4}	3
1.3		3
2	S(1|2|3|4) (R(1|2|3|4)){6} T(1|2|3|4)	2
3.1		2
3.2	Da die Breite beliebig, aber größer als 0 ist, müsste sich der Automat die Differenz der / und \ merken. Da diese aber beliebig groß werden kann, bräuchte der Automat einen Speicher oder unendlich viele Zustände. Da beides nicht möglich ist, kann ein endlicher Automat Spitzen mit beliebiger Breite nicht erkennen.	2

<table>
<tr><th>Aufg.</th><th>erwartete Leistungen</th><th>BE</th></tr>
<tr><td>4.1</td><td> __
| |
 /   \
/   \
\   /
/   \
| |
\   /
 __/</td><td>1</td></tr>
<tr><td>4.2</td><td>Das Syntaxdiagramm enthält nur die Terminalsymbole, die in Regel 1 vorgegeben sind, Rumpf beginnt und endet mit einem _. Das Zeichen _ kommt an keiner anderen Stelle in den Syntaxdiagrammen vor. Innenteil muss mindestens ein | oder ein / gefolgt von einem \ beinhalten. Die Regel 1 ist somit erfüllt.
Im Syntaxdiagramm Innenteil ist das verengende Terminal \ nur nach vorherigem Terminal / möglich. Somit ist Regel 2 erfüllt.
Das Syntaxdiagramm Innenteil stellt zudem sicher, dass / und \ nur paarweise vorkommen. Somit ist auch Regel 3 erfüllt.

Durch die rekursiven Aufrufe von „Innenteil“ können die Eingaben ansonsten beliebig groß werden. Die Syntaxdiagramme beschreiben damit alle Wörter der Sprache R_u.</td><td>3</td></tr>
<tr><td>4.3</td><td>N = {Rumpfteil, Innenteil}
T = { _ , /, |, \ }
Startsymbol = Rumpfteil
P = {Rumpfteil → _ Innenteil _
Innenteil → | ∨ /\ ∨ / Innenteil \ ∨ Innenteil Innenteil}</td><td>2</td></tr>
<tr><td>4.4</td><td>Lösung als Ableitungsbaum:

Rumpfteil
– _
– Innenteil
  – Innenteil
    – |
  – Innenteil
    – /
    – Innenteil
      – Innenteil
        – /
        – \
      – Innenteil
        – /
        – Innenteil
          – |
        – \
    – \
– _</td><td></td></tr>
</table>

<table>
<tr><th>Aufg.</th><th>erwartete Leistungen</th><th>BE</th></tr>
<tr><td></td><td>Lösung als Ableitung:
Rumpfteil → _Innenteil_ → _Innenteil Innenteil_
→_|Innenteil_ →_|/Innenteil_ → _|/Innenteil Innenteil_
→_|//\ Innenteil _ → _|//\/Innenteil_ → _|//\/|_</td><td>2</td></tr>
<tr><td>4.5</td><td>Die Syntaxdiagramme zeigen, dass die Sprache R_u durch eine Grammatik erzeugt werden kann, die auf den linken Seiten der Produktionsregeln mit jeweils einem Nichtterminal auskommt. Daher ist die Sprache bzw. die Grammatik mindestens kontextfrei. Da allerdings auf den rechten Seiten der Produktionen sichergestellt werden muss, dass zu jedem / ein \ existiert, muss es Regeln geben, bei denen mehr als ein Terminal auf der rechten Seite steht. Daher kann die Grammatik nicht regulär sein. Somit ist die erzeugende Grammatik und damit auch die Sprache kontextfrei.</td><td>2</td></tr>
<tr><td>5</td><td></td><td></td></tr>
</table>

```
// In die Register R1 bis R3 wird der Wert 0 geladen.
            LOAD #0
            STORE 1
            STORE 2
            STORE 3

// In R4 wird 9 geladen.
            LOAD #9
            STORE 4

// Der Wert aus R4 wird geladen und inkrementiert
// und somit auf das nächste Zeichen gesetzt.
MARKE:      LOAD 4
            ADD #1
            STORE 4
// Der Wert des Registers, auf das R4 zeigt, wird geladen.
            LOAD *4

// Ende der Zeichenkette erreicht: Sprung zur Marke FERTIG
            JZERO FERTIG
            SUB #1
// Liegt "1" vor: Sprung nach EINS
            JZERO EINS
            SUB #1
// Liegt "2" vor: Sprung zur Marke ZWEI
            JZERO ZWEI
            SUB #1
// Liegt "3" vor: Sprung zur Marke DREI
            JZERO DREI
// Andere Zeichen liefern einen Fehler
            GOTO FEHLER

EINS:       LOAD #1
            STORE 1
            GOTO MARKE

ZWEI:       LOAD 2
// Wert des Registers R2 wird um 1 inkrementiert
            ADD #1
            STORE 2
            GOTO MARKE
```

Nicht für den Prüfling bestimmt

Aufg.	erwartete Leistungen	BE

	<pre>DREI: LOAD 3
	// Wert des Registers R3 wird um 1 inkrementiert
	            ADD #1
	            STORE 3
	// Prüfung, ob mindestens gleich viele "2" wie "3" vorliegen
	             SUB 2
	// Mehr "3" als "2": Fehler
	            JNZERO FEHLER
	// sonst Sprung zu MARKE
	            GOTO MARKE
	
	// Prüfung, ob "2" vorkommt
	FERTIG: LOAD 2
	// Wenn nicht, prüfen, ob "1" vorkommt
	            JZERO EINSEN
	// Sonst prüfen, ob gleich viele "2" und "3" vorkommen
	            SUB 3
	            JNZERO FEHLER
	            GOTO ENDE
	EINSEN: LOAD 1
	            JZERO FEHLER
	            LOAD #0
	            GOTO ENDE
	FEHLER: LOAD #1
	ENDE: END</pre>
		7

	Summe	**30**

III Bewertung und Beurteilung

Die Bewertung und Beurteilung erfolgt unter Beachtung der nachfolgenden Vorgaben nach § 33 der Oberstufen- und Abiturverordnung (OAVO) in der jeweils geltenden Fassung. Bei der Bewertung und Beurteilung der sprachlichen Richtigkeit in der deutschen Sprache sind die Bestimmungen des § 9 Abs. 12 Satz 3 OAVO in Verbindung mit Anlage 9b anzuwenden.

Bei der Bewertung und Beurteilung der Übersetzungsleistung in den Fächern Latein und Altgriechisch sind die Bestimmungen des § 9 Abs. 14 OAVO in Verbindung mit Anlage 9c anzuwenden.

Der Fehlerindex ist nach Anlage 9b zu § 9 Abs. 12 OAVO zu berechnen. Für die Ermittlung der Punkte nach Anlage 9a zu § 9 Abs. 12 OAVO sowie Anlage 9c zu § 9 Abs. 14 OAVO wird jeweils der ganzzahlige nicht gerundete Prozentsatz bzw. Fehlerindex zugrunde gelegt.

Für die Bewertung in den modernen Fremdsprachen ist der „Erlass zur Bewertung und Beurteilung von schriftlichen Arbeiten in allen Grund- und Leistungskursen der neu beginnenden und fortgeführten modernen Fremdsprachen in der gymnasialen Oberstufe, dem beruflichen Gymnasium, dem Abendgymnasium und dem Hessenkolleg“ vom 7. August 2020 (ABl. S. 519) zugrunde zu legen. Demnach erfolgt die Bewertung und Beurteilung mit der Maßgabe, dass lediglich bei der Ermittlung des Prüfungsergebnisses (Note) aus Prüfungsteil 1 und 2 gerundet wird.

Darüber hinaus sind die Vorgaben der Erlasse „Hinweise zur Vorbereitung auf die schriftlichen Abiturprüfungen (Abiturerlass)“ und „Durchführungsbestimmungen zum Landesabitur“ in der für den Abiturjahrgang geltenden Fassung zu beachten.

Im Fach Informatik (Leistungskurs) werden Vorschläge zu den Themen der drei Kurshalbjahre Q1 (Algorithmik und objektorientierte Modellierung), Q2 (Datenbanken) und Q3 (Konzepte und Anwendungen der theoretischen Informatik) vorgelegt, wobei die Prüfungsleistung aus der Bearbeitung je eines Vorschlags zu jedem Halbjahresthema besteht, wofür insgesamt maximal 100 BE vergeben werden können. Ein Prüfungsergebnis von **5 Punkten (ausreichend)** setzt voraus, dass insgesamt 45% der zu vergebenden BE erreicht werden. Ein Prüfungsergebnis von **11 Punkten (gut)** setzt voraus, dass insgesamt 75% der zu vergebenden BE erreicht werden.

Gewichtung der Aufgaben und Zuordnung der Bewertungseinheiten zu den Anforderungsbereichen

Aufgabe	**Bewertungseinheiten in den Anforderungsbereichen**			**Summe**
	AFB I	**AFB II**	**AFB III**	
1	4	3		**7**
2		2		**2**
3		3	1	**4**
4	5	3	2	**10**
5		4	3	**7**
Summe	**9**	**15**	**6**	**30**

Die auf die Anforderungsbereiche verteilten Bewertungseinheiten innerhalb der Aufgaben sind als Richtwerte zu verstehen.